Development of Collective Control Architectures for Small Quadruped Robots Based on Human Swarming Behavior

Daniel W. Palmer[1], Marc Kirschenbaum[1], Jon Murton[1],
Ravi Vaidyanathan[2*], Roger D. Quinn[2]

1. Department of Mathematics and Computer Science, John Carroll University, University Heights, OH, U.S.A.
2. Department of Mechanical and Aerospace Engineering, Case Western Reserve University, Cleveland, OH, U.S.A.

Abstract

We introduce a method of designing behaviors for swarms of micro-robots based on observation of human beings executing various tasks collectively. As a case study, we have focused on the development of decentralized control strategies specifically applicable to swarms of the *Mini-Whegs* quadruped robot. The design process consists of carefully mapping hardware requirements for the robotic platforms in question, then tasking large groups (swarms) of human beings to perform mission specific tasks within the constraints of the robotic vehicle. A basic software engine has been developed and implemented to support on-line human swarm experiments in a virtual environment, with subsequent off-line algorithm extraction following for eventual transfer onto robotic platforms. In our ongoing work, a variety of virtual robotic swarm experiments have been performed, and various methods of algorithm extraction explored. Beyond swarm controller development, one of the most useful and practical aspects of this technology is that it enables those involved in micro-robotic research to understand from a first hand perspective the issues involved in performing global tasks with limited sensor information. We believe that the mining of virtual human swarm behaviors will lead to the successful development of control architectures capable of directing microrobot swarms in the field, as well as provide insights into the social behavior of all manner of multi-agent systems.

Keywords: Microrobots, Swarm-Based Control, Collective Control, Biologically-Inspired, Human Swarming

1. Introduction

Highly mobile small vehicles, sometimes called micro-robots, are better suited for certain missions than larger vehicles. For example, they can aid in search and rescue because their diminutive size enables them to fit into tight spaces, such as those found in rubble and in caves. As another example, a group of small robots provide robustness through redundancy for remote missions such as extraterrestrial exploration. Mobile small robots

[*] Corresponding author, rxv@po.cwru.edu

are also appropriate for insect inspired research because their scale is similar to that of the insect models.

Achieving effective use of micro-robots requires more than just controlling a single vehicle; the larger problem, with a potentially greater payoff, resides in establishing coordinated behavior of groups (or *swarms*) of such vehicles. At present, however, there remains a noticeable gap between the development and simulation of swarm control strategies, and their implementation on robotic hardware platforms for field-use.

In this work the development of decentralized control strategies specifically applicable to swarms of the *Mini-Whegs* [1] quadruped robot is investigated. As a method of controller inspiration, we are examining the swarm behavior of large groups of human beings working under conditions analogous to that of the autonomous robots. By conducting multiple observations of human swarms performing constrained experiments, patterns of successful behavior emerge. Using these patterns, swarm algorithms can be reconstructed and applied to control micro-robotic platforms. We believe that the mining of human-based swarm behaviors conducted under realistic hardware constraints will lead to the successful development of control architectures capable of directing micro-robot swarms in the field.

2. Micro-robot Control Parameters

In order for a control strategy to achieve field-viability, it is critical that it be designed to function within constraints analogous to those imposed by hardware platforms. This is especially true in the micro-robotic arena, where several limitations (power and mobility in particular), are very strict. An architecture relying on any level of global communication, for example, is unlikely to be effective under these restrictions.

Collective control strategies developed in this work are designed with the highly mobile small robots called “Mini-Whegs” in mind [1]. They are derived from the larger Whegs series of robots, which benefit from abstracted cockroach locomotion principles [2]. Key to their success is the three spoke appendages, called “whegs”, which combine the speed and simplicity of wheels with the climbing mobility of legs. To be more compact than the larger Whegs vehicles, Mini-Whegs uses four whegs phased in an alternating diagonal gait. These 9 cm long robots can run at sustained speeds of over 10 body lengths per second and run over obstacles that are taller than their leg length. They can run forward and backward, either right-side up or up-side down. Their robust construction allows them to tumble down a flight of stairs with no damage and carry a payload equal to twice their weight. A jumping mechanism has also been developed that enables Mini-Whegs to surmount much larger obstacles such as stair steps. **Figure 1** shows a photograph of the Mini-Whegs IV robot.

In the human swarm controller development environment, each “agent” in the human swarm is considered to be a single Mini-Whegs platform. Their mobility and power are constrained with the capabilities of the robot itself. Furthermore, their communication and sensing capabilities are also restricted to those currently possible on a small robot such as mini-whegs. Given these limitations the agents are then tasked to interact with one-another in mission scenarios to achieve emergence.

3. Human Swarm Controller Inspiration

A circular dependency exists at the core of designing and building autonomous robots that can work together in swarms. Before developing swarm control algorithms in simulation, it is necessary to know the robot's capabilities, yet before committing to a particular robot configuration, designers must be certain of its sufficiency to support a collective control strategy. Of critical importance in this process is understanding the relationship between the low-level local information-based behavior of individual agents and the high-level global actions of the swarm.

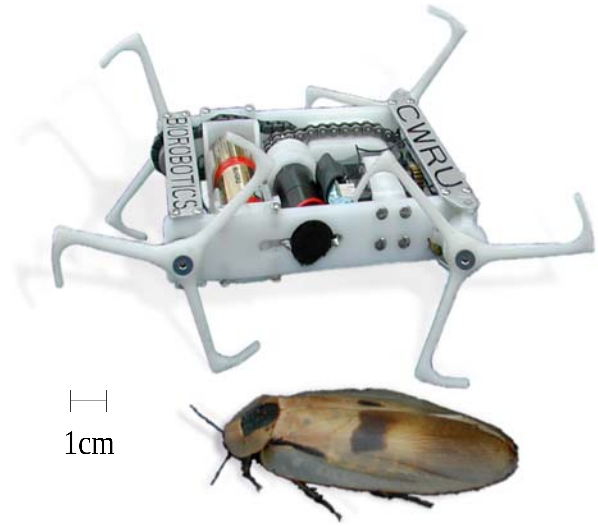

Figure 1. *Photograph showing relative sizes of the Mini-Whegs IV robot and a Blaberus gigantius cockroach.*

One escape route from this dilemma is an iterative, brute force approach that estimates the robot capabilities and approximates a starting point for a control strategy. When this does not work, successive refinement of both sets of parameters is necessary. While it is easy to modify or upgrade the virtual specs of a proposed robot, developing corresponding autonomous collective control strategies has proven to be very difficult, time-consuming and costly. If the controller does not accomplish the intended goal, extensive testing can give clues as to what changes to make, but the process is iterative and lengthy. Ultimately the approach may not be viable without changing the robot's capabilities, which can dictate re-developing the controller. Even if the control strategy succeeds, this is just one data point, a task-specific control strategy that works with a particular robot configuration. The goal is to develop a general purpose robot, therefore, the entire process must be repeated for many different scenarios to determine a viable working set of robot sensors and communication features.

We introduce a novel alternative approach in this work. Instead of developing many autonomous, experiment-specific, robot-specific controllers, we use *human beings* as the computational foundation of simulated robot swarms. The primary benefit of this architecture is the flexibility and adaptability of the humans, allowing us to consider the widest range of possibilities with the least amount of development effort.

In our past work, we have demonstrated the general usefulness of leveraging human computation, in which up to 100 people were enlisted to work together, directly interacting in a large, enclosed space to accomplish a wide variety of collective tasks [3,4]. After recording the actions taken by human swarms, we have reverse-engineered the observed algorithms and successfully applied them to simulated robotic swarms.

Human flexibility and adaptability is a strength we want to leverage, but it is also an obstacle. Humans have sophisticated sensors and communication skills that can impede the reverse-engineering process. For example, hand waving, body language, facial expressions and other sometimes subtle cues all convey meaning and propagate information that cannot be easily captured by our observations or easily reproduced in simulation. To tailor the human reverse-engineering process for robots with restrictions in power, sensors and communication abilities, the data-collecting bonanza of the human visual processing system must be limited. We accomplish this by physically separating the humans and creating a virtual human swarm. Connected remotely over a network, humans take in only the data provided by simulated sensors, they only interact with each other over constrained communication channels and they can only perform simulated actions that mirror the abilities of a proposed robot design.

4. Swarm Algorithm Development

This mix of unlimited human flexibility and adaptability constrained by realistic robotic limitations allows the accomplishment of several objectives in comparison to alternative controller development strategies: (1) it reduces the time, effort and cost of simulating and evaluating different scenarios; (2) it provides for the extraction of control algorithms by reverse-engineering the strategies employed by the humans; (3) it can be used to determine the minimum onboard sensor and communication requirements necessary for many different applications; and (4) it provides for evaluating the autonomous implementations of the extracted algorithms once they are implemented. We examine each of these objectives in turn.

4.1. Reduction of time and cost in development

Leveraging the capacity of the human brain for logical reasoning, common sense and flexibility, dramatically reduces the time and cost to produce control strategies by eliminating the need for physical robot collectives or sophisticated software simulations. Humans, interacting over networked computers, can drive robots within a swarm based on the robot's local information. The network enforces the limited sensor and communication abilities of the robots, allowing the humans to interactively produce control strategies that do not exceed their capabilities. This provides agents with the cognitive ability to understand natural language and can therefore be quickly *programmed* to operate within the constraints of a particular scenario.

One of the powerful features of the virtual swarm is that it provides quick feedback since many different scenarios can be tried and evaluated in a small amount of time. Humans are capable of taking a description of an articulated goal and producing appropriate low-level actions. Thus, many configurations of sensors, battery power, bandwidth, and communication reliability can be easily investigated. To do this without a human "driver" requires either building autonomous intelligence into a simulation system, which is costly, and very time consuming, or building an actual swarm of physical robots which is much more expensive, much more time-consuming. This also severs the feedback loop between robot design and control strategy development.

4.2. Algorithm extraction

The controller strategy will be developed in a four-step process. First, a virtual swarm is used with a simulated robot. Then, once the virtual swarm is able to successfully accomplish the intended goal of the experiment, it is necessary to mine an algorithm to understand how the humans solved the problem. Third, the human controller must be replaced with a software controller, and finally, the simulated robots must be replaced with their physical counterparts. In this section we focus on step two: algorithm mining.

A prototype system for implementing virtual human swarms is currently under development. The distributed system controls what a human at a remote keyboard sees and hears, as well as, restricting and monitoring their communication (see Figure 2B). For example, the software can enforce reduced sensor capabilities, prevent global communication, and limit possible agent actions. The virtual swarm software can record all actions and communications during an experiment making it easier to reverse-engineer control strategies for the robot collective. By repeatedly replaying the recorded data, successful patterns of control become apparent. Once the developers have a good understanding of what the controller must do, it is then appropriate to expend the considerable effort and energy to encapsulate these behaviors in simple rule sets. The complexity of this task is made simpler by the fact that now the developers have a better understanding of how the swarm control strategy should behave. The human-centric system provides information in a more useful order than if the algorithms had to be developed first and then tested.

4.3. Sensor and communication design

A primary goal is to produce robots as cheaply as possible. To this end, the algorithm mining should try to produce a solution that uses a minimal set of sensors and communication facilities.

Using humans as the computational intelligence of simulated robots provides an "upper bound" of the capability of a swarm of those robots. Clearly, for a given configuration of sensors, power, and communication, if human-level reasoning capabilities cannot produce a working solution, then an autonomous controller is unattainable.

There is a complex relationship between the robot's sensor capabilities, the robustness of its communication (both in bandwidth and reliability) and the complexity of the tasks it can complete. A long-range goal of this research is to explore and describe this inter-dependency. The fast feedback of the human-centric simulation platform is the only realistic way to collect enough data to see successful patterns in the solution space.

4.4 Testing and evaluation

The development of the human-centric system produces a customized testbed for the controller software as a natural consequence. The commands from the humans are simply replaced by ones generated by a controller. The rest of the system remains intact and global observation of performance of the autonomous controllers can be directly compared with their human counterparts. This again provides quick and useful feedback to help fine-tune and verify the controller software. This architecture also supports humans working with autonomously controlled agents. Such a hybrid swarm supports further testing of the extracted control algorithms and helps transition to a purely autonomous system. Earlier in the algorithm development process, a hybrid swarm can also allow experimentation with larger swarms than there are human drivers. In this scenario, the autonomous controllers will be given a default set of behaviors, which can be overridden by directives from human controlled agents.

5. Virtual Human Swarm (VHS) System

Figure 2A shows a typical scene from a physical human swarm experiment. In these experiments the collection of 100 humans are given descriptions of a physical goal and as a group, work towards reaching it. (In this example, agents were instructed to organize by color and form lines.) A major problem encountered in physical human swarm experiments is the difficulty in controlling most aspects of the experiment's environment. When the humans are in each other's physical presence, they can shout across the room, copy behaviors and read visual cues.

Figure 2A: Human Swarm Experiment Snapshot

Figure 2B: Virtual Human Swarm Experiment Snapshot

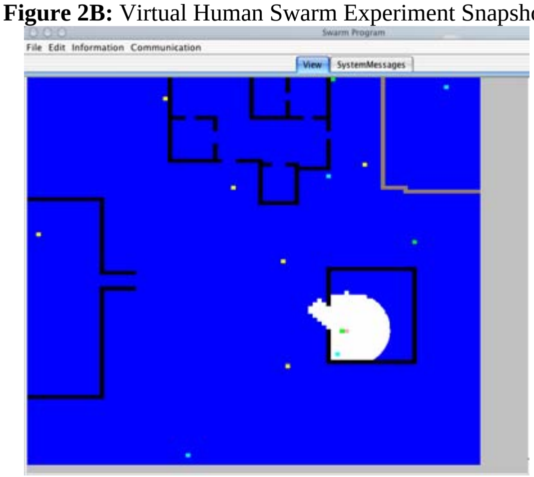

Figure 2C: Screenshot of Virtual Swarm Server

In **Figure 2B**, five humans control agents in a small virtual swarm. In this example, all participants were in the same room; however this is not a requirement. In actual practice, the participants will be isolated from each other. **Figure 2C** shows a snapshot of the Virtual Human Swarm (VHS) server displaying a global view of the environment. The solid black lines indicate impregnable walls, the gray lines in the upper right corner indicate obstacles, which can be traversed but at a slower rate, and the white area is the sensor region for the beacon located at the center. The human agents only see a restricted portion of this overall environment.

Examples of the agent views are presented in **Figures 2D-2F**. The unbalanced hourglass shape shows the forward and backward sensing region corresponding to that of a mini-wheg. The greater forward sensing capabilities are represented by the larger triangular area. The humans interact with the environment through buttons below the viewing area. Six buttons correspond to the six different movements that the mini-whegs can perform. (Forward, Backward, and since the mini-whegs can not make lateral movements the other moves are Forward-left, Forward-right, Back-left, and Back-right.) The remaining button, Ping, allows the agent to locate other agents in their vicinity but outside their immediate sensor range through alternate measures such as sonar or radio. As evidenced in **Figure 2D,** there also is the possibility of communicating with other

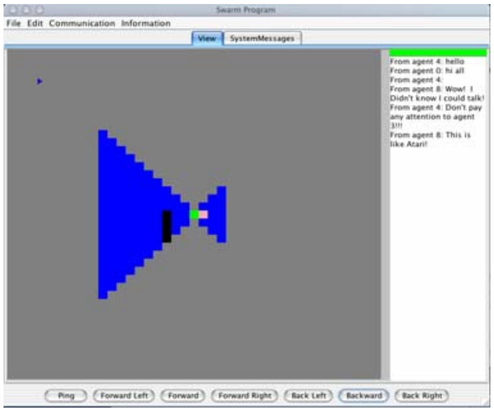

Figure 2D: Screenshot of Virtual Swarm Agent 1

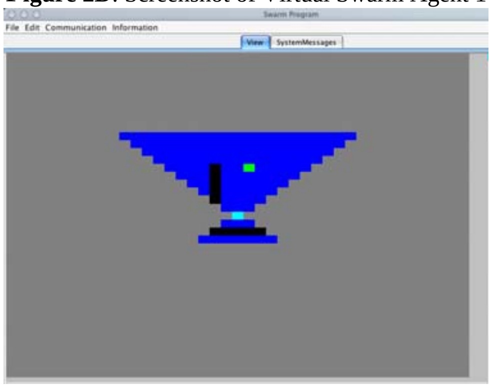

Figure 2E: Screenshot of Virtual Swarm Agent 2

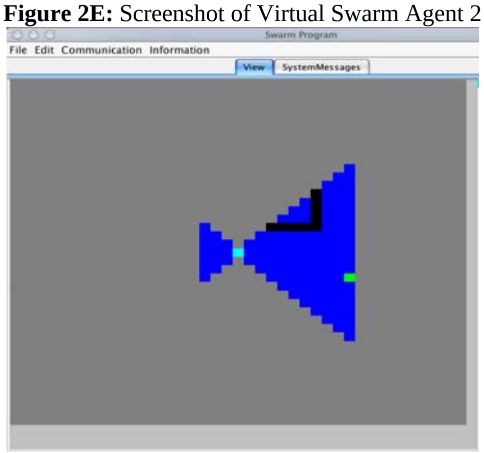

Figure 2F: Screenshot of Virtual Swarm Agent 3

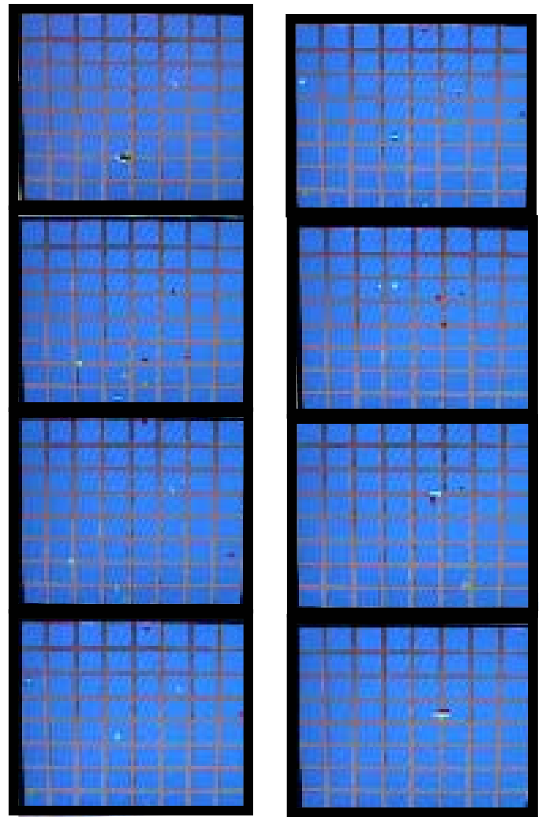

Figure 3A: Screenshots of Dispersion (left) scenario
Figure 3B: Screenshots of Congregation (right) scenario

agents either in broadcast or point to point mode. The current implementation allows full chatting capability but this will be restricted once we start collecting data. **Figure 2D** shows the agent right next to the beacon. **Figure 2E** and **Figure 2F** show two other agents' views. Each has another agent within its sensor field. These are indicated by the color square within the triangular sensor region.

Although it is premature to present results from the research, we give a flavor of the kinds of data we expect to collect. Initially, we have several simple scenarios designed to determine the ability of humans to complete tasks in the VHS. Scenario one starts with all agents centrally located and the task is to disperse to a uniform density. Scenario two is the reverse: from a uniformly distributed starting point, all agents must congregate. No collecting point for the swarm is specified. **Figures 3A** and **3B** show the general nature of the data we expect to collect. In the left column, from top to bottom, the agents can be seen starting out clustered in the lower portion of the screen, and spreading out to fill the region. The right column, starting at the top fully spread out shows the agents converging together. Note that they regroup at a different location than where they started.

Other scenarios include a “Find-the-beacon” exercise, Formation-formation and establishing a perimeter about a given location. **Figures 2C** through **2F** show the overview and three agent views of a “find-the-beacon” exercise. In **Figure 2C**, the beacon can be scene inside the square in the lower right side of the screen. The circular region surrounding the beacon is a visual representation of the sensing region of the beacon. Notice that it is a line-of-sight device, so it cannot be sensed through walls and only directionally out the doorway.

As of the writing of this paper, the software is still under development. The screenshots presented here were collected during system testing. We expect to complete the system and begin running experiments shortly. We expect to collect and analyze data for presentation at the workshop.

6. Future Work

One of the most appealing aspects of this research is the ease of “programming/reprogramming” a virtual human swarm. High level commands are given and away they go. Eventually, if we make the interface as good as what can be seen in video games, there is the possibility of having an endless number of swarm agents to perform our experiments. Our job will be to create virtual worlds that capture the environments our mini-whegs will face and to be creative enough to keep our human agents from losing interest. This could be a cost effective method for the initial design of a robotic controller. There are already plans to build a 3D version of the software that will run continuously over the internet. Given a suitable structure of points and scoring opportunities, we may find that thousands of undergraduates are willing to contribute to this research. We also envision the possibility of mounting micro-cameras on the mini-whegs and having human drive the actual hardware

Acknowledgements

The authors gratefully acknowledge Ms. Kelly Zajac for her software development efforts in the virtual swarm implementation.

References

1. Morrey, J.M., Lambrecht, B., Horchler, A.D., Ritzmann, R.E., and Quinn, R.D., (2003) “Highly Mobile and Robust Small Quadruped Robots” To appear in *IEEE Int. Conf. On Intelligent Robots and Systems* (IROS’03), Los Vegas, Nevada, 2003.
2. Quinn, R.D., Kingsley, D.A., Offi, J.T. and Ritzmann, R.E., (2002), “Improved Mobility through Abstracted Biological Principles”, *IEEE Int. Conf. On Intelligent Robots and Systems (IROS’02)*, Lausanne, Switzerland, 2002.
3. Palmer, D.W., Kirshembaum, M., Kovacina, M.A., Murton, J.P., Zajac, K.M., “Self-referential Biological Inspiration: Humans Observing Human Swarms to Identify Swarm Programming Techniques” *7th World Multi-Conference on Systemics, Cybernetics & Informatics (SCI)* Orlando, FL, 2003.
4. Dan Palmer, Marc Kirschenbaum, Jon P. Murton, Michael A. Kovacina, Danile H. Steinberg, Sam N. Calabrese, Kelly M. Zajac, Chad M. Hantak, Jason E. Schatz, “Using a Collection of Humans as an Execution Testbed for Swarm Algorithms”, Proceedings of the IEEE Swarm Intelligence Symposium, pages 58 – 64, , April 24-26, 2003.